**Создание backend части информационной системы**

Создать REST сервер при помощи springframework. В проекте должна быть реализована трехслойная архитектура.

Основные пакеты проекта:

* config – конфигурация контейнера
* entity – сущности данной предметной области. Должны иметь jpa маппинг с базой данных.
* repository – доступ к базе данных. Реализуется при помощи SpringData.
* service – бизнес-логика приложения. Использует repository для реализации бизнес-логики приложения. Помимо реализаций обязательно должны присутствовать интерфейсы.
* controller – непосредственно API сервера. Обработка и ответ на запросы клиентов.

При создании сервера можно использовать как серверный springframework, так и spring boot. В проекте должна быть реализована система аутентификации на основе http basic auth.

Помимо стандартных CRUD методов в контроллерах должны быть представлены методы бизнес-логики.

К проекту должен быть приложен скрипт создания базы данных.

**Индивидуальное задание**

|  |  |
| --- | --- |
| **№ варианта** | **Задание** |
| 1,6,11 | **Информационная система библиотеки.** Информационная система библиотеки позволяет искать книги в своем каталоге[guest], учитывать выдачу книг на абонемент и возврат книг[manager], а также позволяет добавлять книги в фонд и списывать их[admin]. |
| 2,7,12 | **Информационная система поликлиники.** Информационная система поликлиники позволяет ставить и снимать больных с учета[manager], записывать больных на прием к врачам[manager], учитывать факт приема[manager], позволяет вести историю болезни (медицинскую карту) больного [manager], добавлять и удалять врачей[admin] и просматривать список врачей[guest]. |
| 3,8,13 | **Информационная система деканата.** Информационная система деканата позволяет принимать и отчислять студентов[manager], вести учет успеваемости по итогам сессии[manager], переводить студентов из группы в группу и с курса на курс[admin], просматривать студентов группы[guest]. |
| 4,9,14 | **Система учета рабочего времени.** Система учета рабочего времени позволяет руководителям выдавать задания и отслеживать ход их выполнения[admin], а исполнителям — вести учет рабочего времени[manager], затраченного на выполнение каждого задания, а также узнать выданные сотрудникам задания [guest] |
| 5,10,15 | **Информационная система технической экспертизы.** Информационная система технической экспертизы позволяет соискателям грантов подавать заявки[guest], независимым экспертам оценивать заявки[manager], а держателям фонда принимать решение о выдаче грантов по результатам экспертизы заявок[admin]. |

В квадратных скобках указаны роли, которые имеют доступ к указанным возможностям бизнес-логики.

* guest – любой пользователь, в том числе неаутентифицированный.
* manager – пользователь с ограниченными правами использования системы
* admin – администартор системы с полными правами

Методы CRUD (добавления, удаления, редактирования и просмотра сущностей) должны быть доступны только администратору, если условием не указано иное.

**Все взаимодействие с сервером производится при помощи REST-протокола (рекомендуется использовать postman)**

**Примечание:** данная работа будет применена в финальной лабораторной работе по Angular.